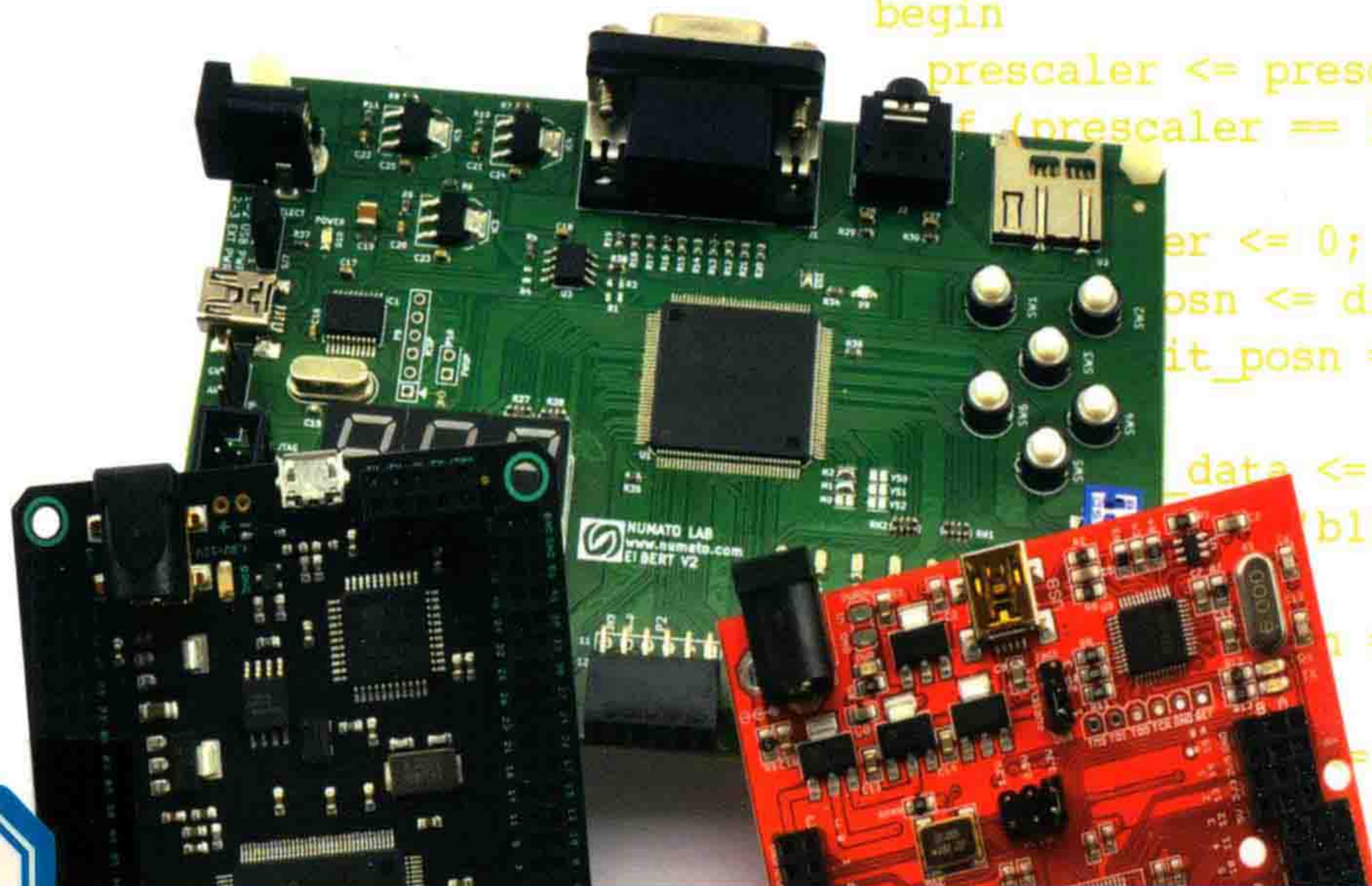


FPGA编程从零开始

使用Verilog

[美] 西蒙·蒙克(Simon Monk) 著
李 杨 别志松 译



单板机、微处理器设计、电子电路等类别的畅销书

向开始使用Verilog和Xilinx FPGA的所有人士强烈推荐本书！有这本优秀书籍做指导，我的Numato Lab Mimas V2 Spartan 6 FPGA开发板很快就能运行了。

为满足一部分入门级读者的需要，本书简明扼要地介绍一些关于逻辑门的知识。Verilog是硬件描述语言，你可以像使用C等高级编程语言一样，用Verilog来描述逻辑，编写FPGA程序。

——亚马逊书店读者

清华大学出版社

FPGA编程从零开始

使用Verilog

[美] 西蒙·蒙克(Simon Monk) 著

李杨 别志松 译

清华大学出版社

北 京

Simon Monk

Programming FPGAs: Getting Started with Verilog

EISBN: 978-1-25-964376-7

Copyright © 2017 by McGraw-Hill Education.

All Rights reserved. No part of this publication may be reproduced or transmitted in any form or by any means, electronic or mechanical, including without limitation photocopying, recording, taping, or any database, information or retrieval system, without the prior written permission of the publisher.

This authorized Chinese translation edition is jointly published by McGraw-Hill Education and Tsinghua University Press Limited. This edition is authorized for sale in the People's Republic of China only, excluding Hong Kong, Macao SAR and Taiwan.

Translation copyright © 2018 by McGraw-Hill Education and Tsinghua University Press Limited.

版权所有。未经出版人事先书面许可, 对本出版物的任何部分不得以任何方式或途径复制或传播, 包括但不限于复印、录制、录音, 或通过任何数据库、信息或可检索的系统。

本授权中文简体字翻译版由麦格劳-希尔(亚洲)教育出版公司和清华大学出版社有限公司合作出版。

此版本经授权仅限在中国大陆地区销售、不能销往中国香港、澳门特别行政区和中国台湾地区。

版权©2018 由麦格劳-希尔(亚洲)教育出版公司与清华大学出版社有限公司所有。

北京市版权局著作权合同登记号 图字: 01-2018-0334

本书封面贴有 McGraw-Hill Education 公司防伪标签, 无标签者不得销售。

版权所有, 侵权必究。侵权举报电话: 010-62782989 13701121933

图书在版编目(CIP)数据

FPGA编程从零开始 使用Verilog/(美)西蒙·蒙克(Simon Monk)著; 李杨, 别志松译. —北京: 清华大学出版社, 2018

书名原文: Programming FPGAs: Getting Started with Verilog

ISBN 978-7-302-50134-3

I. ①F… II. ①西… ②李… ③别… III. ④可编程序逻辑器件—系统设计 IV. ④TP332.1

中国版本图书馆 CIP 数据核字(2018)第 112346 号

责任编辑: 王 军 韩宏志

装帧设计: 孔祥峰

责任校对: 曹 阳

责任印制: 宋 林

出版发行: 清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址: 北京清华大学学研大厦 A 座

邮 编: 100084

社总机: 010-62770175

邮 购: 010-62786544

投稿与读者服务: 010-62776969, c-service@tup.tsinghua.edu.cn

质 量 反 馈: 010-62772015, zhiliang@tup.tsinghua.edu.cn

印 装 者: 北京嘉实印刷有限公司

经 销: 全国新华书店

开 本: 148mm×210mm

印 张: 6.125

字 数: 153 千字

版 次: 2018 年 7 月第 1 版

印 次: 2018 年 7 月第 1 次印刷

定 价: 49.80 元

产品编号: 078173-01

译者序

Verilog 是最流行的硬件描述语言之一，已被 IEEE 列为标准硬件描述语言。其基本语法与 C 语言相近，这一设计初衷使其对于数字电路研发者更容易理解和上手。

本书作者 Simon Monk 博士是数字电路工程设计领域的专家，撰写了多本有关开源硬件方面的著作。本书是 Simon Monk 博士的经典之作，对于使用 Verilog 语言进行 FPGA 开发的初学者来说，是一本绝佳的入门书籍。虽然对于大部分初学者来说，数字电路设计晦涩难懂，硬件编程和调试也难以入门，但本书由浅入深、循序渐进，将复杂的主题讲得简单易懂，将枯燥的技术讲得活灵活现；从基本的数字电路概念讲起，全面介绍使用 Verilog 进行 FPGA 编程的重要环节，指导你开始使用 Mojo、Papilio One 和 Elbert 2 这三种流行的 FPGA 开发板。本书虽然简短，但麻雀虽小，五脏俱全，你将从头到尾为大量工程编写指令，包括 LED 译码器、计时器、单音生成器，甚至是存储器映射的视频显示器。作者在该领域经验丰富，提供了多个紧贴实用的开发实例。相信读者在学习完本书后，能快速设计出自己的第一个作品！何乐而不为呢？

在这里要感谢清华大学出版社的编辑，他们对本书的翻译和出版投入了很多心血，保证了本书顺利付梓。本书全部章节由李杨、别志松翻译，参与翻译的还有王迎、王辞等人。

对于本书，译者在翻译过程中始终保持谨慎的态度，但难免存在疏漏。如有任何意见和建议，恳请广大读者批评指正。

作者简介

Simon Monk 拥有控制和计算机科学学士学位，以及软件工程博士学位。Simon 是一名全职作家，迄今已撰写多本书籍，包括 *Programming Arduino*、*Programming Raspberry Pi* 和 *Hacking Electronics*，并参与撰写 *Practical Electronics for Inventors*。Simon 的个人网站是 MonkMakes.com，Twitter 是 @simonmonk2。

致 谢

一如既往地感谢 Linda 的耐心和支持。

感谢 TAB/McGraw-Hill 和 MPS Limited 的 Michael McCabe、Patty Wallenburg 和他们的同事。一如往常，与如此卓越的团队合作非常愉快。

同时感谢 Duncan Amos 有益而全面的技术审阅。

前 言

用自己的芯片完成想做的事，岂不是一件乐事？当然，现场可编程门阵列(Field-Programmable Gate Array, FPGA)可让你非常接近这一梦想。FPGA 并非为你专门设计的芯片，而是通用芯片，能被配置用来完成你希望做的任何事情。

此外，要配置 FPGA，既可绘制原理图，也可使用硬件定义语言 Verilog；如果你的设计是成功的，Verilog 也能用于生产真正的定制芯片。尽管本书也将展示如何使用原理图编辑器进行设计，但本书的重点是指导你学习 Verilog 语言。

可根据自己的需要多次修改 FPGA 配置，使其成为原型化设计的优秀工具。如果设计问题浮出水面，你可对设备重新编程，直到消除所有漏洞为止。当你意识到可真正配置 FPGA 来包含能运行程序的处理器时，这种十分出奇的灵活性就会显现出来。

在本书中，你将学习 FPGA 的一般使用原则，将学习本书描述的示例，并在三种最流行的 FPGA 评估板(Mojo、Papilio One 和 Elbert 2)上运行这些示例。

尽管从逻辑上讲，微控制器可胜任 FPGA 能完成的大部分工作，但 FPGA 的运行速度更快；另外，一些人员发现，相对于实现复杂的算法，描述逻辑门和硬件更简单。你可使用 FPGA 实现微控制器或其他处理器(以及其他人的工作)。

在其中一种低成本 FPGA 开发板上使用 Verilog 编程，可能最令人信服的原因仅在于学习一些新知识，收获一些乐趣！

读者可访问<https://github.com/simonmonk/pro-fpgas>下载本书各章的项目文件，也可扫描封底的二维码下载。

目 录

第 1 章	逻辑	1
1.1	逻辑门.....	1
1.1.1	非门	2
1.1.2	与门	3
1.1.3	或门	3
1.1.4	与非门和或非门.....	4
1.1.5	异或门	5
1.2	二进制.....	6
1.3	添加逻辑.....	8
1.4	触发器.....	9
1.5	移位寄存器.....	11
1.6	二进制计数器	12
1.7	小结.....	13
第 2 章	FPGA.....	15
2.1	FPGA 的工作原理.....	15
2.2	Elbert 2	17
2.3	Mojo.....	18
2.4	Papilio	20
2.5	软件设置.....	22
2.5.1	安装 ISE	22
2.5.2	安装 Elbert 软件	24

2.5.3	安装 Mojo 软件	25
2.5.4	安装 Papilio 软件	26
2.6	项目文件	26
2.7	小结	27
第 3 章	绘制逻辑	29
3.1	数据选择器示例	29
3.1.1	步骤 1: 创建一个新项目	30
3.1.2	步骤 2: 创建一个新的原理图	34
3.1.3	步骤 3: 添加逻辑符号	36
3.1.4	步骤 4: 连接门	36
3.1.5	步骤 5: 添加 IO 标记	37
3.1.6	步骤 6: 创建用户约束文件	38
3.1.7	步骤 7: 生成 .bit 文件	42
3.1.8	步骤 8: 编写开发板	44
3.1.9	测试结果	46
3.2	一个 4 位计数器示例	48
3.2.1	绘制原理图	49
3.2.2	实现约束文件	49
3.2.3	测试计数器	52
3.3	小结	52
第 4 章	Verilog 简介	53
4.1	模块	53
4.2	引线、寄存器和总线	54
4.3	并行执行	54
4.4	数字格式	54
4.5	使用 Verilog 编写的数据选择器	55
4.6	使用 Verilog 编写的计数器	59
4.7	同步逻辑	62

4.8	小结.....	62
第 5 章	模块化 Verilog	63
5.1	七段译码器.....	63
5.2	按钮去抖.....	68
5.3	复用七段显示器和计数器.....	73
5.3.1	项目结构	74
5.3.2	display_7_seg.....	76
5.3.3	counter_7_seg	79
5.3.4	用户约束文件.....	81
5.3.5	导入模块源代码.....	82
5.3.6	设置顶层模块.....	82
5.3.7	3 数位版本.....	83
5.3.8	测试.....	83
5.4	小结.....	84
第 6 章	计时器示例.....	85
6.1	状态机.....	85
6.2	状态机设计.....	87
6.3	硬件.....	88
6.3.1	你之所需	88
6.3.2	构建.....	88
6.4	模块.....	90
6.5	用户约束文件	91
6.6	计时器模块.....	92
6.6.1	输入和输出.....	92
6.6.2	按压按钮	92
6.6.3	报警器实例.....	93
6.6.4	建模时间和显示.....	93
6.6.5	状态机实现.....	94

6.6.6	任务	96
6.7	测试.....	98
6.8	小结.....	98
第 7 章	PWM 和伺服电机	99
7.1	脉冲宽度调制	99
7.2	PWM 模块	100
7.2.1	PWM 模块输入和输出.....	101
7.2.2	PWM 测试模块	101
7.2.3	试一试	104
7.3	伺服电机.....	104
7.4	硬件.....	105
7.4.1	你之所需	105
7.4.2	构建	106
7.5	伺服模块.....	109
7.6	小结.....	112
第 8 章	音频	113
8.1	单音生成.....	113
8.2	Mojo 的音频输出.....	115
8.3	通用音/频发生器	116
8.3.1	单音模块	116
8.3.2	tone_tester 模块	118
8.3.3	测试	118
8.4	播放音频文件	121
8.4.1	音频文件	121
8.4.2	RAM.....	122
8.4.3	wav_player 模块	122
8.4.4	测试	125
8.4.5	准备自己的音频.....	125

8.5 小结.....	128
第 9 章 视频	129
9.1 VGA.....	129
9.2 VGA 定时同步	132
9.3 绘制矩形.....	133
9.3.1 VGA 模块.....	134
9.3.2 VGA 和 Elbert 2	137
9.4 使物体运动.....	138
9.5 存储器映射显示	141
9.6 小结.....	143
第 10 章 扩展内容	145
10.1 仿真.....	145
10.2 更深层次的内容	146
10.3 核和软处理器	147
10.4 更多 Papilio 内容	147
10.5 更多 Mojo 内容.....	149
10.6 小结.....	150
附录 A 资源	151
附录 B Elbert 2 参考	155
附录 C Mojo 参考	165
附录 D Papilio One 参考.....	173

第 1 章

逻 辑

现场可编程门阵列(FPGA)是依赖数字逻辑的数字设备。计算机硬件使用数字逻辑。每一次计算、渲染到显示屏上的每一个像素，以及音轨中的每一个音符都可使用数字逻辑的构件来创建。

虽然相对于物理电子器件，数字逻辑有时更像一个抽象的数学概念，但数字逻辑的逻辑门和其他部件是由晶体管蚀刻到集成电路(Integrated Circuit, IC)上进行构建的。在 FPGA 中，通过绘制逻辑门来设计电路，然后将逻辑门映射到 FPGA 上的通用门并连接在一起，以实现逻辑设计。另外，也可使用 Verilog 或其他硬件描述语言来描述逻辑。

你可购买包含少量逻辑门的芯片，例如具有四个两输入 NAND 门的 7400 芯片。然而，这些芯片实际上仅用于维护使用它们的旧系统，或用于教学用途。

1.1 逻辑门

逻辑门具有输入和输出。这些数字输入和输出可以是高电平或低电平。低数字输入或输出由接近 0V(接地)的电压表征。高数字输入通常超过逻辑电源电压的一半，高数字输出是正电源电压。FPGA 电源电压通常为 1.8V、3.3V 或 5V，大部分 FPGA 可运行于一段电压范围

内，有些允许在一台设备上使用多个逻辑电压。

逻辑门的描述较为复杂，因为逻辑门的名称(非、与、或等)在英文中也有对应的含义。为避免混淆，本书将门名称大写。

1.1.1 非门

最简单的逻辑门是非(NOT)门，有时称为反相器。它具有单输入和单输出。如果输入是高电平，则输出是低电平；反之亦然。图 1-1 显示了非门的原理图符号。真值表列出每种可能的输入组合及对应输出。按照惯例，输入命名时，使用字母表开头的字母，如 A、B 和 C。输出命名时，通常是 Q 或临近字母表结尾处的字母，如 X、Y 和 Z。

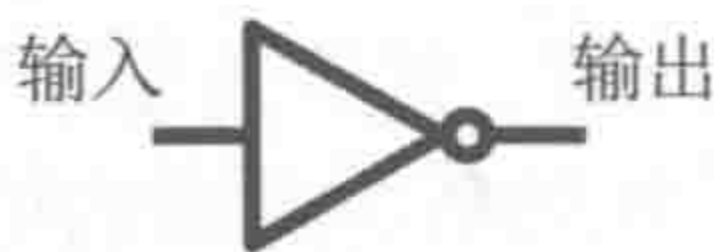


图 1-1 非门

为描述逻辑门或一组逻辑门的行为，可使用真值表。真值表定义逻辑为每个可能的输入或输出组合提供的输出。非门的真值表如表 1-1 所示。字母 H 和 L 或数字 1 和 0 用于代替高(high)和低(low)。

表 1-1 非门的真值表

输入	输出
L	H
H	L

如果将一个非门的输出连接至第二个非门，如图 1-2 所示，组合电路的输出一定为输入本身。

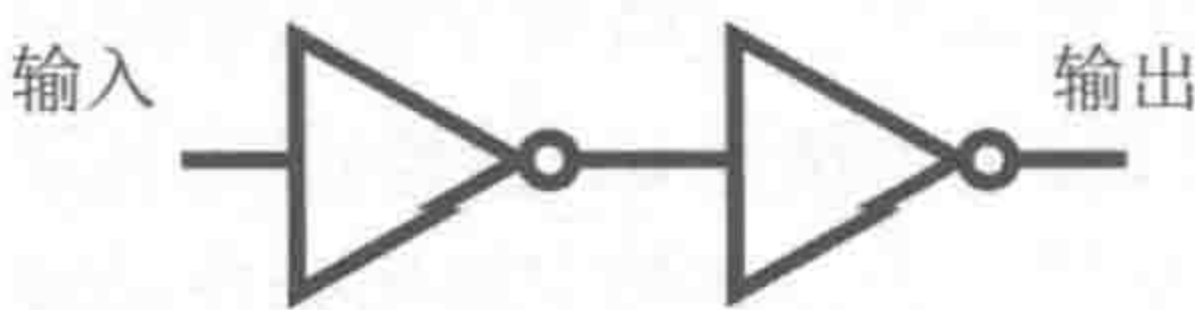


图 1-2 两个非门

1.1.2 与门

顾名思义，与门的输出仅在其所有输入都为高电平时为高。图 1-3 显示了两输入与门的符号，表 1-2 是与门的真值表。

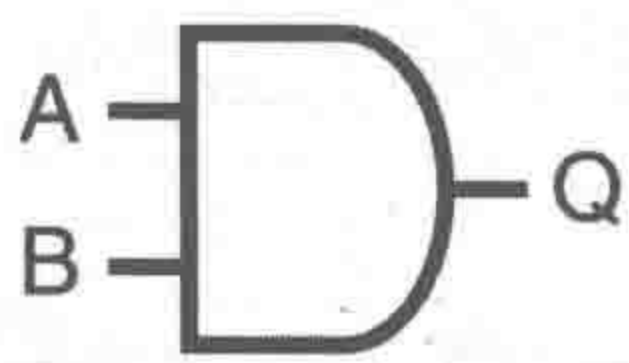


图 1-3 与门

表 1-2 与门的真值表

输入 A	输入 B	输出 Q
L	L	L
L	H	L
H	L	L
H	H	H

1.1.3 或门

顾名思义，任一输入为高电平时，或门的输出为高。图 1-4 显示了两输入或门的符号，表 1-3 是或门的真值表。

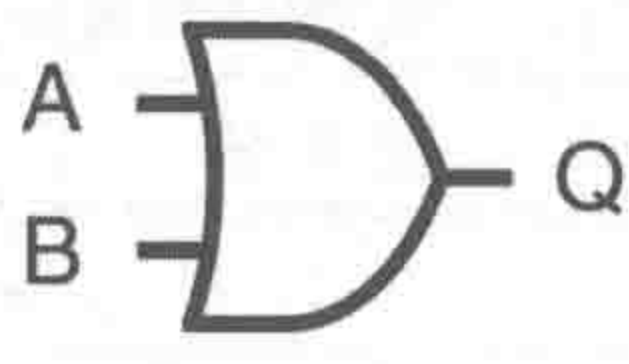


图 1-4 或门

表 1-3 或门的真值表

输入 A	输入 B	输出 Q
L	L	L
L	H	H
H	L	H
H	H	H

1.1.4 与非门和或非门

图 1-1 所示的非门输出的小电路表明了门的反相功能。与非门(NOT AND, NAND)是与门的输出反相, 而或非门(NOT OR, NOR)是或门的输出反相。图 1-5 显示了这两个门的符号, 表 1-4 和表 1-5 分别是这两个门的真值表。

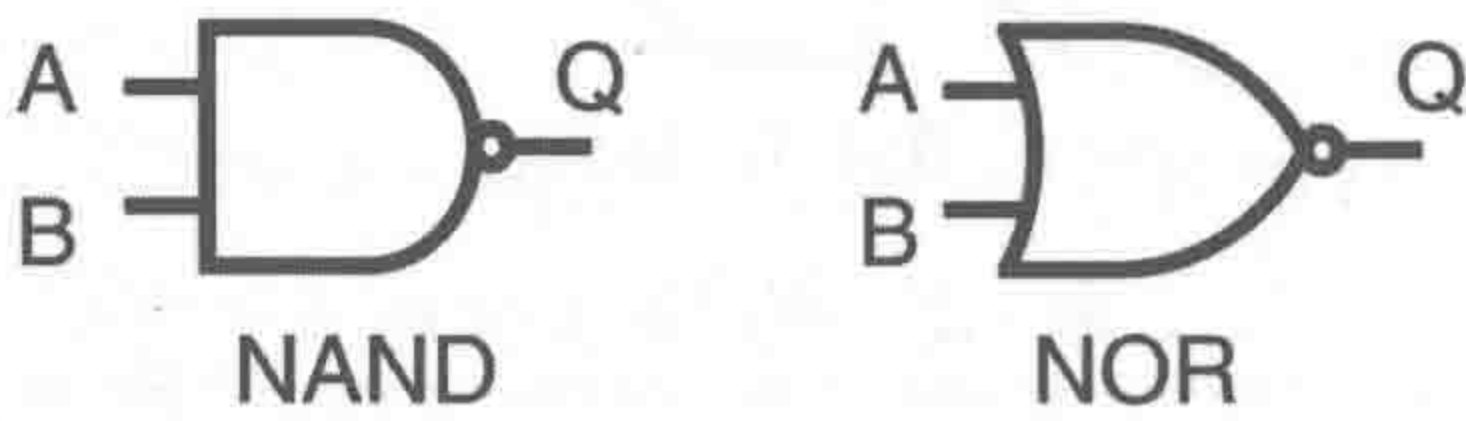


图 1-5 与非门和或非门

表 1-4 与非门的真值表

输入 A	输入 B	输出 Q
L	L	H
L	H	H
H	L	H
H	H	L

表 1-5 或非门的真值表

输入 A	输入 B	输出 Q
L	L	H
L	H	L
H	L	L
H	H	L

与非门和或非门都作为通用门(universal gate)来描述, 因为通过对输入和输出的反相, 二者均可作为其他类型的门使用。另外, 可组合一个与非门和一个或门的输入形成一个非门。例如, 图 1-6 显示了如